International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 09 | Sep 2020 www.irjet.net p-ISSN: 2395-0072

Crop Price prediction using Random Forest and Decision Tree Regression

S Brunda[1], Nimish L[2], Chiranthan S[2], Arbaaz Khan[2]

[1]Assistant Professor, Department of Computer Science and Engineering, JSS Science and Technology University Mysuru, India
[2]Student, Department of Computer Science and Engineering, JSS Science and Technology University Mysuru, India

---

Abstract - *Machine Learning with the Prediction model has gained its popularity through its promising results. Its application has been incorporated in this paper too where various regression model has been studied to predict the Crop prices. The crop price prediction assist the farmers to plan their next crop to be grown and avoid hyperinflation. The dataset has 330 different crops altogether. Different models have been investigated for their performance and compared. The Results shows that the Random Forest Regression and Decision Tree Regressor has the best prediction model among all with an accuracy of around 99%.*

Key Words: Decision tree, Prediction, Super- vised Machine Learning, Random Forest Regression, Hyperparameter tuning.

1. INTRODUCTION

India is an agriculture-based country and farmer community is the backbone of the agriculture sector. APMC's are the place where the farmer expects a good price for his yield. This paper aims to predict the prices of Agricultural commodities using Machine Learning and get the overview of the crop stocks anywhere at the given time. Such generated knowledge helps farmer and APMC in their decision making and thereby achieving their goal of making profits and maintain a balance between deficit and surplus productions. The prices of agricultural commodity have an unstable nature which may rise or fall differently causing negative effect on the economy. The work completed here for predicting prices of agricultural commodities is helpful for the farmers as they can sow crops depending upon the prediction of future cost. Farming items have regular rates; these rates are spread over the whole year. In the event that these rates are known to farmers ahead of time, it will be guarantying on Rate on Investments (ROI). Agricultural specialists can pursue these charts and anticipate advertise rates which can be informed to farmers.

1.1 Background

The prediction results of the Random Forest Regressor and Decision tree Regressor has been observed for the current dataset. Random forests are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest.

2. LITERATURE SURVEY

Some crop prices were forecasted from the data in Taiwan market [8]. Author develops a crop price forecasting service based on the market prices published by the Council of Agriculture. The developed service was integrated in the smart agri-management platform, providing an interface for historical price retrieval and future price forecast. Daily updated prices covered 15 different markets and more than 100 different crops, and there are four algorithms ARIMA, PLS, ANN, and RSMPLS available for price forecasting. According to the experiment for price forecasting, the recommended algorithms are PLS and ANN. This paper suggests different algorithms for different crops and there is no single algorithm which predict the price for all the crops.

3. DATA SET & DATA VISUALIZATION

The data has been collected from the Open Government Data(OGD) India [4]. The data consists of the 330 different Crop which are grown throughout India and the Prices are taken as the mean of the Prices which were available in different places. The dataset consists of the Crop prices from the year 2011 to 2018. This raw data required data cleaning for the redundant and null values. The dataset totally consists of 8,68,966 entries with 4 attributes consisting of Index, Item Name, Date and Price. After cleaning the data. The data type of the Item Name is of string, Prices in int, Date in DD/MM/YYYY format. All three features are a predictive features which could be used for the prediction model. The given historic data is one of the distinguishing feature that would be assisting the model in the price prediction.

As a part of this machine learning problem, the first step is to gather the data and perform the feature engineering. For attaining a better accuracy in the price prediction, it is required to tune the hyperparameter of the model. This paper will focus on the optimizing the Decision Tree Regressor using Scikit tools [1]. In any Machine learning model, the model parameters such as slope and intercept refer to the hyperparameters which has to be set before training the model. In the case of Decision Tree Regressor, the splitting upon the nodes which are consumed by the model directly depends upon the hyperparameters set [5]. Scikit-Learning has default values for the models, but these are not guaranteed to be optimal for all problems.

Data visualization and Proof of Concept(POC):

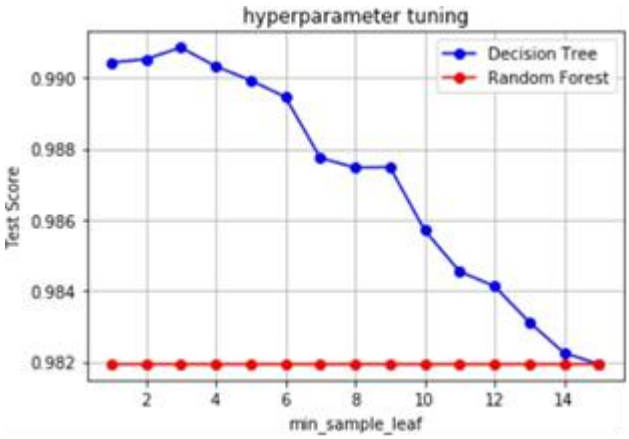

Fig. 1: Determining the optimal value of min sample leaf

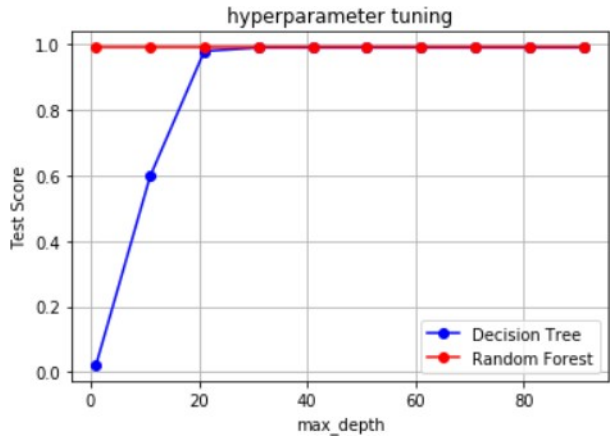

Fig. 2: Determining the optimal value of max depth

In order to avoid the overfitting, the dataset is split into the training and testing set with 80% data for training and remaining 20% data for testing. The data is split with the some random state so that the training is not biased.

Here are some of the different models applied on the given dataset for the price prediction:

3.1. Decision Tree Regressor

A decision tree is a supervised machine learning model used to predict a target by learning decision rules from features. They are recursively constructed starting from the root node till the child nodes. The attributes of the data and the model hyperparameters together assists in the node construction. Some of the hyperparameters in the DecisionTreeRegressor which are considered are:

- min sample leaf = The minimum number of samples required to be at a leaf node
- n estimators = number of trees in the forest
- max features = max number of features considered for splitting a node
- max depth = max number of levels in each decision tree

In accordance with the values of the dataset, Fig 1 depicts how the model behaves well when the minimum number of samples required to be at a leaf node is set to 3. The model deteriorates with the increase in the value of min sample leaf. Here, the optimal value for the min sample leaf is taken as 3. In the case of max depth, Fig 2 shows that the model trains well linearly up to the value 20 after which there is further improvements. Hence, the max depth is set to 20. Likewise, other hyperparameters are set to their default value and could be even tuned if necessary. To much tuning the hyperparameters would lead to overfitting. After training the model, the Decision Tree Regressor has attained an accuracy of 99.08%. The total time taken for training the Decision Tree Regressor model is 1.67s. Although Decision Trees seems to fit well for the data, they are sensitive to the specific data on which they have been trained. With the subsequent addition of dataset and distinguishing attributes to the model, the predictions would be quite different.

3.2. Random forest Regressor

The trees in the Random Forest Regressor run in parallel. While the model is building different trees there is no interaction between them. Random for- est being meta-estimator aggregates many decision trees output as a result at the end. The behavior of the Random Forest is quite different from that of the Decision Tree Regressor as observed in the Fig 3. The variation of the hyperparameters does not affect the model and it attains a constant score of 0.9917 with slight variation in 6th or 7th decimal places. The total time taken for training the Random Forest Regressor is 1.77s. The scatter plot in the Fig 3 has been plotted for only 30% of the test cases for the better visualizing the data.

Various other models has been tried out and the Score for the test set are recorded in the Table I

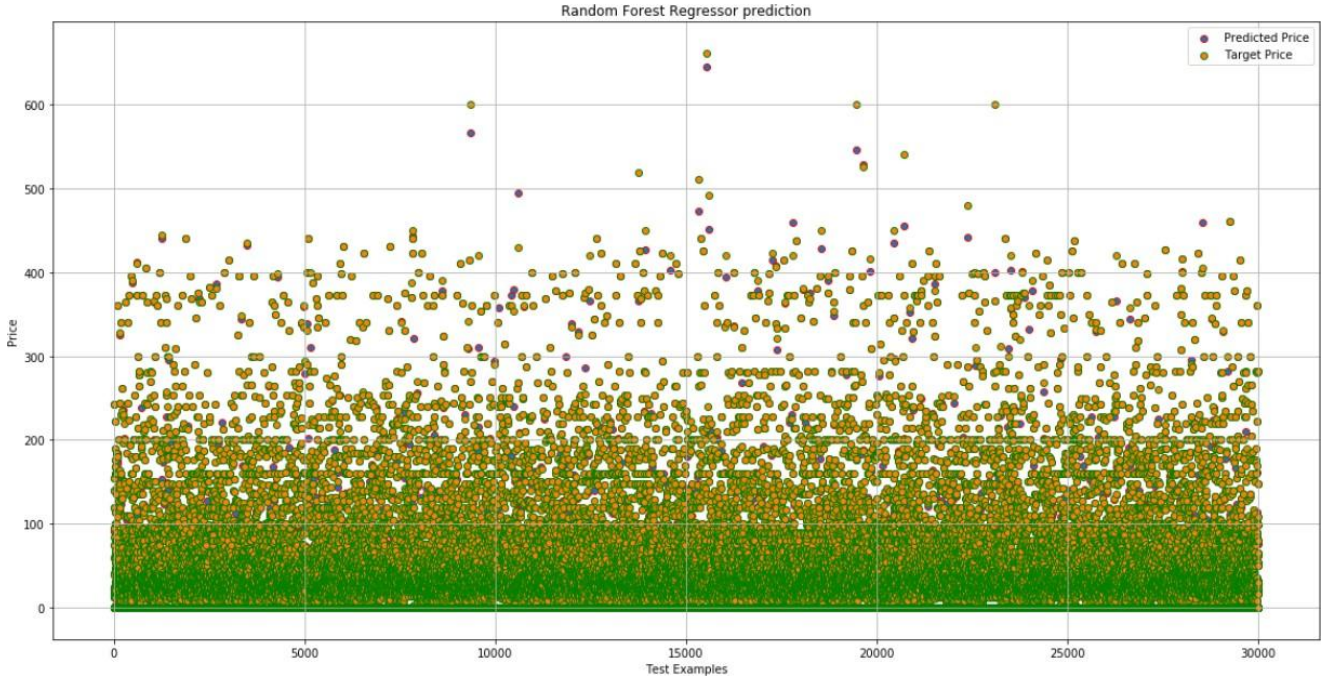

Fig. 3: Random Forest Prediction for test cases

4. CONCLUSIONS

The main objective is to predict the crop price and indirectly estimate the profit for the given crops before sowing. The training datasets so obtained provide the enough insights for predicting the appropriate price and demand in the markets. With any additions to the dataset, Random forest Regressor would not be requiring much of the hyperparameter tuning compared to the Decision tree Regressor. In this sense, Random Forests are an effective tool in this prediction. The strength of the individual predictors and their correlations gives insight into the ability of the random forest to predict. Thus, the model would assist the farmer to take the right decision in choosing the crops to be grown.

Model Name	Score on test Set
Simple Linear Regressor	0.0223197270
Linear SVR	-0.0739817537
Random Forest Regressor	0.9917790951
Decision Tree Regressor	0.9908507658
K Nearest Neighbor	0.3439384563
Lasso	0.0223197220
Ridge	0.0223197270
Adaboost Regressor	0.4284328182
Isotonic Regressor	-0.504892965

Table 1: Table showing 8 models and their Prediction

REFERENCES

[1] scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html

[2] en.wikipedia.org/wiki/Featurescaling#Standardization

[3] scikit-learn.org/stable/autoexamples/models election/plotlearningcurve.html

[4] https://data.gov.in/catalog/variety-wise-daily-market-prices-data-other-vegetable?filters%5Bfieldcatalog reference%5D=93651&format=json&offs et=0&limit=6&sort%5Bcreated%5D=desc

[5] https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74

[6] Y. Peng, C. Hsu and P. Huang, "Developing crop price forecasting service using open data from Taiwan markets," 2015 Conference on Technologies and Applications of Artificial Intelligence (TAAI), Tainan, 2015, pp. 172-175, doi: 10.1109/TAAI.2015.7407108.

[7] https://towardsdatascience.com/random-forest-and-its-implementation-71824ced454f.

[8] https://www.stat.berkeley.edu/breiman/random forest2001.pdf